SIAM J. SCI. COMPUT.
Vol. 29, No. 2, pp. 459–480

STRING GRADIENT WEIGHTED MOVING FINITE ELEMENTS IN MULTIPLE DIMENSIONS WITH APPLICATIONS IN TWO DIMENSIONS*

A. WACHER† AND I. SOBEY‡

Abstract. We formulate the string gradient weighted moving finite element method (SGWMFE) for systems of PDEs in multiple dimensions. Then we illustrate implementation issues for the method using two dimensions. The method is applied successfully to solve the highly nonlinear unsteady porous medium equation in two dimensions and the results are compared to exact solutions. We proceed to present results for the method when applied to the Gray Scott chemical reaction diffusion model in two dimensions. Finally, we apply the method to the shallow water equations in two dimensions and compare the meshes produced with those produced using gradient weighted finite element methods (GWMFE). Our conclusion is that SGWMFE is an easily applied member of the group of moving finite element methods.

Key words. moving finite elements, mesh adaptation

AMS subject classifications. 74S05, 65M60, 65M50

DOI. 10.1137/040619557

1. Introduction.

This paper has two objectives. The first is to present the general formulation for a string gradient weighted moving finite element method (SGWMFE) in multiple space dimensions. The second is to demonstrate the implementation of SGWMFE in several nonlinear systems of PDEs, the porous medium equation, the Gray Scott equations, and the shallow water equations. Solutions of each problem can be found using other methods, and we are not suggesting or attempting to demonstrate that SGWMFE is better or worse than other moving (or nonmoving) finite element or discretization methods; rather, it is an implementation of gradient weighted moving finite elements (GWMFE) with an elegant general form that is straightforward to use.

Moving finite element methods (MFE), originally introduced in [14], are well established to provide reliable solutions to unsteady systems of PDEs, often with less computational cost than fixed mesh methods. There is a wide literature dealing with moving mesh methods, both for finite elements and for finite differences. The text of [1] gives an extensive view of MFE and [5, 11, 15, 16] describe recent applications of moving mesh methods. GWMFE were developed in detail in [8] and [9], and the string version, suggested in [12], was implemented in [17]. For other background on moving mesh methods, see [17], where there is an extended reference list on moving mesh methods including MFE, GWMFE, and moving finite difference type methods. Here our interest is however in developing details of SGWMFE for systems of PDEs in two or more dimensions. Using a projection matrix, a very general form of the method can be developed and details are illustrated for a system of two PDEs. There

*Received by the editors November 24, 2004; accepted for publication (in revised form) August 30, 2006; published electronically March 15, 2007.
http://www.siam.org/journals/sisc/29-2/61955.html

†Faculty of Aerospace Engineering, Technion City, Haifa 32000, Israel (abigail.wacher@comlab.ox.ac.uk). This author was supported in part by the National Science and Engineering Research Council of Canada in the form of the NSERC graduate scholarship and by Corpus Christi College, Oxford.

‡Oxford University Computing Laboratory, Parks Road, Oxford OX1 3PQ, UK (ian.sobey@comlab.ox.ac.uk).

are a number of implementation issues for MFE and the details of implementation of SGWMFE are examined for a number of commonly occurring terms in first and second order PDEs.

In the remainder of this section we set out the three test problems that are considered in this paper. In the second section we develop a mathematical formulation for SGWMFE that is elegant and easily generalized to arbitrary systems of evolutionary PDEs in multiple space dimensions. The third section considers implementation issues for a system two independent variables in two space dimensions, focusing the theory from section two for this case and showing details of how different terms are treated. The fourth section gives solutions for the three test problems using SGWMFE.

1.1. Porous medium equation. The first test problem we consider is the application to the porous medium equation in two space dimensions. While this is a single PDE rather than a system, it admits an exact similarity solution under some initial conditions so that the results of applying SGWMFE can be compared to the exact solution.

The porous medium equation is well known (see [6] or [4]) but in computational terms represents a hard problem because under some parameter values, the solution develops a singularity in the slope of an advancing front. The porous medium equation in two dimensions is given by

$$u_t = \nabla \cdot (u^m \nabla u), \tag{1.1}$$

and we consider the problem with $u = 0$ imposed at the boundary.

This is a problem with a known set of similarity solutions when the initial data has circular symmetry. The symmetric self-similar solutions used are of the same form as in [2] and originally from [4]. Earlier studies on self-similar numerical solutions of the porous medium equation can be found in [6] and [7]. The following form represents the solutions for the one- ($d = 1$), two- ($d = 2$), and three- ($d = 3$) dimensional problems for an initial distribution centered at the origin and starting at time t_0:

$$u(r,t) = \left\{ \begin{array}{ll} \dfrac{1}{\lambda^d(t)}\left(1 - \left(\dfrac{r}{r_0\lambda(t)}\right)^2\right)^{1/m}, & r \le r_0\lambda(t) \\ 0, & r > r_0\lambda(t) \end{array} \right\}, \tag{1.2}$$

where r is distance from the origin, m is an integer greater than 0, and

$$\lambda(t) = \left(\frac{t}{t_0}\right)^{\frac{1}{2+md}} \quad \text{with} \quad t_0 = \frac{r_0^2 m}{2(2+md)} \ . \tag{1.3}$$

The computations described below are for times $T = t - t_0 > 0$. The computational difficulty with this problem is that the slope of the solution at the moving front, $r = r_0\lambda(t)$, is singular when $m > 1$.

1.2. Gray Scott equations. The Gray Scott equations are a model for chemical species undergoing reaction and diffusion. Reaction and diffusion of chemical species can lead to a variety of patterns which have beautiful and interesting solutions. For a description of the physics of the model studied here, see [10]. The reader is also referred to [20] for a comparison of results for the following problem using a different class of moving mesh method.

The chemical reaction we consider is that given by

$$\begin{aligned} U + 2V &\to 3V, \\ V &\to P, \end{aligned}$$

where U, V, and P are chemical species. The PDEs that model this process are

$$\begin{aligned} u_t &= r_u \nabla^2 u - uv^2 + f(1-u), \\ v_t &= r_v \nabla^2 v + uv^2 - (f+k)v, \end{aligned} \tag{1.4}$$

where u and v represent their concentrations, r_u and r_v are their diffusion rates, k represents the rate of conversion of V to P, and f represents the rate of the process that feeds U and drains U, V, and P.

The initial conditions considered are block functions for each of the chemical concentrations over the domain $[0,1] \times [0,1]$:

$$u(x,y,t) = \left\{ \begin{array}{ll} 0.5, & 0.3 \le (x,y) \le 0.7 \\ 1 & \text{elsewhere} \end{array} \right\}, \tag{1.5}$$

$$v(x,y,t) = \left\{ \begin{array}{ll} 0.25, & 0.3 \le (x,y) \le 0.7 \\ 0 & \text{elsewhere} \end{array} \right\}. \tag{1.6}$$

The coefficients chosen for this problem are $r_u = 8(10^{-5})$, $r_v = 4(10^{-5})$, $f = 0.024$, and $k = 0.06$ with Dirichlet boundary conditions prescribed on all four boundaries of the domain.

1.3. Shallow water equations. The nonlinear shallow water equations provide a further model problem to test SGWMFE. It is also possible to compare the results with those obtained from the GWMFE code used in [8]. In our shallow water problem, a "hump" of stationary water is released at time zero in the center of a square box. As the hump subsides under gravity, a wave propagates away from the center, collides with the reflective boundaries, and propagates back to the center of the box. As they travel, a wave front steepens, forming a shock in the absence of viscosity. In this problem a simulation should capture the shape of the wave front, the wave height, and the front speed.

The nonlinear shallow water equations studied here are, in nondimensional terms, given by

$$\begin{aligned} u_t &= -\nabla \cdot \mathbf{f} + \epsilon \nabla^2 u, \\ v_t &= -\nabla \cdot \mathbf{g} + \epsilon \nabla^2 v, \\ w_t &= -\nabla \cdot \mathbf{h} + \epsilon \nabla^2 w. \end{aligned} \tag{1.7}$$

The physical domain is given by $0 \le x \le 5$, $0 \le y \le 5$, time $t \ge 0$, u is the height of the fluid from the flat bottom, v is the fluid momentum in the x direction, and w is the fluid momentum in the y direction.

The vector functions in (1.7) are

$$\mathbf{f} = (v, w), \quad \mathbf{g} = \left(\frac{v^2}{u} + \frac{u^2}{2}, \frac{vw}{u} \right), \quad \mathbf{h} = \left(\frac{vw}{u}, \frac{w^2}{u} + \frac{u^2}{2} \right). \tag{1.8}$$

The initial conditions assigned are

$$u = 0.2 + e^{-((x-2.5)^2 + (y-2.5)^2)}, \tag{1.9}$$

$$v = 0, \tag{1.10}$$

$$w = 0. \tag{1.11}$$

On the south and north of the domain, the boundary conditions are that $u_y = 0$, $v_y = 0$, and $w = 0$. On the east and west of the domain, the boundary conditions are that $u_x = 0$, $w_x = 0$, and $v = 0$.

2. Formulation of SGWMFE in multiple dimensions. We consider a system of n PDEs in k dimensions with independent variables $x = (x_1, \ldots, x_k)$ and dependent variables $u = (u_1, \ldots, u_n)$ satisfying

$$\frac{\partial u_l}{\partial t} = L_l(u_1, \ldots, u_n), \qquad l = 1, \ldots n, \tag{2.1}$$

where the differential operators L_l are first or second order nonlinear differential operators in space.

The solution u can be considered as an n-dimensional manifold in $(n + k)$-dimensional space; that is,

$$\mathbf{s} = (x_1, \ldots, x_k, u_1, \ldots, u_n)^{\mathrm{T}}. \tag{2.2}$$

Consider a parameterization of the manifold $\xi = (\xi_1, \ldots, \xi_k)$ so that the solution manifold is described by

$$\mathbf{s}(\xi, t) = (x_m(\xi, t), m = 1 \ldots k, u_l(x_j(\xi, t), j = 1 \ldots k, t), l = 1 \ldots n)^{\mathrm{T}}. \tag{2.3}$$

The velocity of the manifold, $\dot{\mathbf{s}}$, is not computed directly; rather, the normal component of the velocity, $\dot{\mathbf{s}}_n$ (not to be confused with the rate of change of the normal to the manifold, so that $\dot{\mathbf{s}}_n \equiv [\dot{\mathbf{s}}]_n$), is calculated, and normal movement of the manifold is deduced from that velocity. Tangential movement of the manifold is not considered since the location of the nodes is not affected.

In order to compute the normal velocity it is necessary to be able to evaluate the normal component of vectors on the manifold. This can be done by defining k tangent vectors to the manifold $\mathbf{X}_m = (0 \ldots 1 \ldots 0, \frac{\partial u_l}{\partial x_m}, l = 1 \ldots n)^{\mathrm{T}}$, $m = 1, \ldots, k$, where the unit value occurs in the mth row. These tangent vectors are always nonparallel and nonzero. Define matrices

$$X = [\mathbf{X}_1, \ldots, \mathbf{X}_k], \quad V = (\mathbf{X}_l^{\mathrm{T}} \mathbf{X}_m) = X^{\mathrm{T}} X,$$

and the properties of the component vectors (nonzero and nonparallel) will guarantee that V^{-1} will exist. Define D by

$$D = \det(V). \tag{2.4}$$

LEMMA 2.1. *The normal* $\mathbf{s}_n$ *to the manifold* $\mathbf{s}$ *is given by*

$$\mathbf{s}_n = P\mathbf{s}, \tag{2.5}$$

where the matrix P *is given by*

$$P = I - XV^{-1}X^{\mathrm{T}}. \tag{2.6}$$

Proof. If $\mathbf{s}_n = \mathbf{s} - \sum_{l=1}^{k} a_l \mathbf{X}_l = \mathbf{s} - X\mathbf{a}$ then choosing $\mathbf{a}$ so that $\mathbf{s}_n \cdot \mathbf{X}_m = 0$ for $m = 1, \ldots, k$ gives $\mathbf{a} = V^{-1}X^{\mathrm{T}}\mathbf{s}$ and hence the equation for P. ∎

Note that $P\mathbf{X}_m = 0$ for $m = 1, \ldots, k$ trivially as the tangent vectors have no normal component.

Having identified the normal component of a vector on the manifold, the normal velocity can be determined. To do this, use

$$\dot{x}_m = \frac{\partial x_m(\xi, t)}{\partial t}, \quad m = 1, \ldots, k, \tag{2.7}$$

and

$$\dot{u}_l = \frac{\partial u_l}{\partial t} + \sum_{j=1}^{k} \frac{\partial u_l}{\partial x_j}\dot{x}_j, \quad l = 1, \ldots, n, \tag{2.8}$$

and rearrangement gives that, if

$$R = \begin{pmatrix} 0_{k\times k} & 0_{k\times n} \\ [-u_{l,m}]_{n\times k} & I_{n\times n} \end{pmatrix}, \tag{2.9}$$

where $0_{k\times n}$ is a k-row, n-column matrix of zeros, $I_{n\times n}$ is an n by n identity matrix, and $[-u_{l,m}]$ is an n-row, k-column matrix with entries $-\frac{\partial u_l}{\partial x_m}$, then

$$R\dot{\mathbf{s}} = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ \frac{\partial u_1}{\partial t} \\ \vdots \\ \frac{\partial u_n}{\partial t} \end{pmatrix}. \tag{2.10}$$

LEMMA 2.2. *The matrices P and R satisfy*

$$PR = P. \tag{2.11}$$

Proof. The matrix R is rewritten as $R = I - [\mathbf{X}_1 \cdots \mathbf{X}_k \mathbf{0} \cdots \mathbf{0}]$, and then we use $P\mathbf{X}_m = 0$. ∎

Thus we obtain

$$\dot{\mathbf{s}}_n = P\dot{\mathbf{s}} = PR\dot{\mathbf{s}} = P\begin{pmatrix} 0 \\ \vdots \\ 0 \\ \frac{\partial u_1}{\partial t} \\ \vdots \\ \frac{\partial u_n}{\partial t} \end{pmatrix} = P\begin{pmatrix} 0 \\ \vdots \\ 0 \\ L_1 \\ \vdots \\ L_n \end{pmatrix} = P\mathbf{L}, \tag{2.12}$$

with $\mathbf{L} = (0, \ldots, 0, L_1, \ldots, L_n)^{\mathrm{T}}$. Thus the normal motion of the manifold is determined by

$$P\big[\dot{\mathbf{s}} - \mathbf{L}\big] = \mathbf{0}. \tag{2.13}$$

This system is central to the solution method since the original n PDEs have been replaced by $n + k$ equations for the manifold velocities $(\dot{x}_1, \ldots, \dot{x}_k, \dot{u}_1, \ldots, \dot{u_n})$ and upon space discretization, these will become $n + k$ nonlinear ODEs in time.

3. Implementation of SGWMFE in two dimensions. Having considered the very general formulation, in this section we look at the details of implementing the method for a system of two PDEs in two dimensions. For simplicity, denote the independent variables x and y and the dependent variables u and v, and let them be governed by

$$u_t = L_1(u, v), \qquad v_t = L_2(u, v). \tag{3.1}$$

The solution (u, v) can be considered as a two-dimensional manifold in four-dimensional space; that is,

$$\mathbf{s} = (x, y, u, v)^{\mathrm{T}}. \tag{3.2}$$

Two tangent vectors to the manifold will be $\mathbf{X}^{\mathrm{T}} = (1, 0, u_x, v_x)$ and $\mathbf{Y}^{\mathrm{T}} = (0, 1, u_y, v_y)$, and the determinant D is

$$D = |\mathbf{X}|^2|\mathbf{Y}|^2 - (\mathbf{X}\cdot\mathbf{Y})^2 = (|\mathbf{X}||\mathbf{Y}||\sin(\theta)|)^2, \tag{3.3}$$

where θ is the angle between $\mathbf{X}$ and $\mathbf{Y}$.

The normal $\mathbf{s}_n$ to the manifold $\mathbf{s}$ is given by

$$\mathbf{s}_n = P\mathbf{s} \tag{3.4}$$

(see (2.5)), where the matrix P from (2.6) is given by

$$P = I - \frac{1}{D}[|\mathbf{Y}|^2(\mathbf{X}\mathbf{X}^T) - (\mathbf{X}\cdot\mathbf{Y})(\mathbf{X}\mathbf{Y}^T) + |\mathbf{X}|^2(\mathbf{Y}\mathbf{Y}^T) - (\mathbf{X}\cdot\mathbf{Y})(\mathbf{Y}\mathbf{X}^T)]. \tag{3.5}$$

The matrix R is

$$R = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -u_x & -u_y & 1 & 0 \\ -v_x & -v_y & 0 & 1 \end{pmatrix} \tag{3.6}$$

so that

$$R\dot{\mathbf{s}} = R\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{u} \\ \dot{v} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ u_t \\ v_t \end{pmatrix}. \tag{3.7}$$

As in the general case, obtain

$$\dot{\mathbf{s}}_n = P\dot{\mathbf{s}} = PR\dot{\mathbf{s}} = P\begin{pmatrix} 0 \\ 0 \\ u_t \\ v_t \end{pmatrix} = P\begin{pmatrix} 0 \\ 0 \\ L_1 \\ L_2 \end{pmatrix} = P\mathbf{L}, \tag{3.8}$$

with $\mathbf{L} = (0, 0, L_1, L_2)^{\mathrm{T}}$. Thus, as before, the normal motion of the manifold is determined by

$$P\big[\dot{\mathbf{s}} - \mathbf{L}\big] = \mathbf{0}. \tag{3.9}$$

Discretization of this system can be cast in a finite element framework by using appropriate test functions—in this case the piecewise linear basis functions which are used also to discretize the solutions for (u, v) at time varying (x, y) nodal locations. If the basis functions are denoted α^i then the discrete equations for the normal motion of the manifold are

$$\int_{\text{manifold}} P(\dot{\mathbf{s}} - \mathbf{L})\alpha^i \mathrm{d}S = \mathbf{0}, \tag{3.10}$$

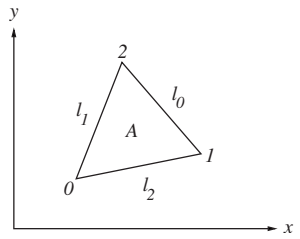

FIGURE 3.1. *Triangle T, showing the orientation of nodes 0, 1, and 2, the projected $x-y$ area A, and the edge lengths ℓ_0, ℓ_1, and ℓ_2.*

where $\mathrm{d}S$ is an element of the manifold and the superscript i ranges over the basis functions. In two dimensions, a triangular cell T can be used, as in Figure 3.1, and then, after piecewise linear discretization, u_x, v_x, u_y, v_y, and hence also P are constant on each cell. The parameterization used for the implementation of SGWMFE is $\xi_1 = x$ and $\xi_2 = y$, so that two infinitesimal vectors $\mathbf{X}\mathrm{d}x$ and $\mathbf{Y}\mathrm{d}y$ span a parallelogram in $4-D$ of area

$$\mathrm{d}S = |\mathbf{X}\mathrm{d}x||\mathbf{Y}\mathrm{d}y||\sin(\theta)| = \sqrt{D}\mathrm{d}x\mathrm{d}y. \tag{3.11}$$

It can also be noted that this SGWMFE formulation is equivalent to a variational formulation which chooses to minimize the functional

$$\psi = \int ||P(\dot{\mathbf{s}} - \mathbf{L})||^2 \mathrm{d}S, \tag{3.12}$$

with respect to the nodal velocities $\dot{\mathbf{s}}_i = (\dot{x}_i, \dot{y}_i, \dot{u}_i, \dot{v}_i)$.

In order to solve the discrete system of ODEs, an algorithm such as the well-known backward differentiation formula 2 (BDF2) found in [3] can be used. The BDF2 integrator is a robust stiff integrator which is necessary for the stiff systems arising in SGWMFE; however, any robust stiff integrator may be used. The BDF2 integration code developed and made available by Carlson and Miller was used for the results in this paper. A detailed outline of the BDF2 integrator routine is in [8].

3.1. Implementation. The implementation of SGWMFE requires some notation for the average value of a scalar or vector valued function f on a cell or an edge. An element of area on the manifold, $\mathrm{d}S$, is related to the projected $x-y$ area by (3.11), and since D is constant on each element, the projected area A of an element with area S is related by $S = \sqrt{D}A$. Hence the average of a function f over a triangular cell T is given by

$$[f]_T = \frac{1}{A}\int_T f\mathrm{d}x\mathrm{d}y = \frac{1}{S}\int_T f\mathrm{d}S. \tag{3.13}$$

We will also assume that the three nodes of the triangle T are labeled $0, 1, 2$, as in Figure 3.1.

The average of f over an edge E of length ℓ is given by

$$[f]_E = \frac{1}{\ell}\int_E f\mathrm{d}s, \tag{3.14}$$

where the integrand is with respect to the infinitesimal length along the edge: $\mathrm{d}s = \sqrt{\mathrm{d}x^2 + \mathrm{d}y^2}$.

3.2. Time derivative terms. As P is constant on an element, the components of the system for time derivatives on a triangular cell T are given by

$$\int_T P\dot{\mathbf{s}}\alpha^0 \mathrm{d}S = P\int_T \dot{\mathbf{s}}\alpha^0 \mathrm{d}S. \tag{3.15}$$

Similar contributions will occur for nodes 1 and 2 of the cell. Since the $\dot{\mathbf{s}}$ and α^0 are linear functions of the two-dimensional parameter ξ, the integrand in (3.15) is quadratic in the components of ξ and hence can be evaluated by the "edge-midpoint rule" which is exact for quadratic functions.

3.3. Flux terms. Since many systems of interest contain a flux term, consider the case where $L_1 = \nabla \cdot \mathbf{f}$. For the given triangular cell T, the contributions from this flux term from the cell onto its 0 node are

$$\int_T P \begin{pmatrix} 0 \\ 0 \\ -\nabla\cdot\mathbf{f} \\ 0 \end{pmatrix} \alpha^0 \mathrm{d}S = \sqrt{D}P \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} \int_T (-\nabla\cdot\mathbf{f})\alpha^0 \mathrm{d}x\mathrm{d}y, \tag{3.16}$$

where, again, the constancy of P and D on each cell enables these to be taken out of the integral. The scalar integral expression may be integrated by parts, as in [9]:

$$\int_T -\nabla\cdot\mathbf{f}\alpha^0 \mathrm{d}x\mathrm{d}y = \int_T \mathbf{f}\cdot\nabla\alpha^0 \mathrm{d}x\mathrm{d}y - \int_{E_1} \alpha^0\mathbf{f}\cdot\widehat{\nu}_1 \mathrm{d}s - \int_{E_2} \alpha^0\mathbf{f}\cdot\widehat{\nu}_2 \mathrm{d}s, \tag{3.17}$$

where E_1 and E_2 are edges adjacent to node 0 and $\widehat{\nu}_1$, $\widehat{\nu}_2$ are outward normals to those edges (α^0 is zero on the third edge).

This component can be simplified by using average values. Defining ℓ_i as the $x-y$ projected lengths of the corresponding edge E_i, (3.17) is rewritten in terms of average values as

$$\int_T -\nabla\cdot\mathbf{f}\alpha^0 \mathrm{d}x\mathrm{d}y = A\nabla\alpha^0\cdot[\mathbf{f}]_T - l_1\widehat{\nu}_1\cdot[\alpha^0\mathbf{f}]_{E_1} - l_2\widehat{\nu}_2\cdot[\alpha^0\mathbf{f}]_{E_2}. \tag{3.18}$$

3.4. Constant coefficient diffusion terms. Another common component of the differential operators L_1 and L_2 is a diffusion term. We consider first the case of constant diffusion; the second case of nonconstant or nonlinear diffusion is considered later. In order to evaluate the integrals of the Laplacian terms, mollification is used. For piecewise linear approximations, the Laplacian is singular at an edge. In order to overcome this, consider the integrals in neighborhoods of each edge and assume that the piecewise linear approximation is the limit of an approximation which can be differentiated twice; that is, the piecewise linear representations of the solution variables need to be mollified so that the first derivatives vary smoothly from cell to cell in a neighborhood whose thickness will be allowed to vanish. The principle here is the same as in the two-dimensional paper of [9]. To do this, some of the same labels as in that paper are used to define cells, edges, endpoints, and unit normal vectors.

Without loss of generality, assume that the 0 to 1 edge E is aligned in the y direction as shown in Figure 3.2. This is done so that the unit outward normal vector to E is $\widehat{\nu} = \widehat{x}$, and the unit tangent vector to E is $\widehat{\tau} = \widehat{y}$. Then on crossing the edge, the values u_y and v_y stay constant, but u_x and v_x are functions of x, which increase by $2a$ and $2b$, respectively, with mean values m_u and m_v. Now assume all three are

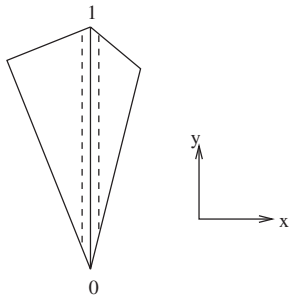

FIGURE 3.2. *Mollification of edge of two triangles.*

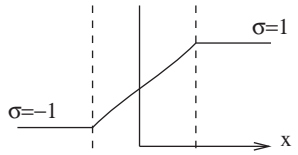

FIGURE 3.3. $\sigma(x)$.

mollified equally; that is, the "**X**" tangent vector to the two-dimensional manifold in four dimensions is

$$\mathbf{X}(x) = \mathbf{M} + \sigma(x)\mathbf{A} = \begin{pmatrix} 1 \\ 0 \\ u_x \\ v_x \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ m_u \\ m_v \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ a \\ b \end{pmatrix} \sigma(x), \tag{3.19}$$

where $\sigma(x)$ varies from -1 to 1 in the neighborhood of the edge as shown in Figure 3.3. Thus the Laplacian contribution is

$$\mathbf{L} \equiv \begin{pmatrix} 0 \\ 0 \\ u_{xx} + u_{yy} \\ v_{xx} + v_{yy} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ a \\ b \end{pmatrix} \sigma'(x). \tag{3.20}$$

The "**Y**" tangent vector to the two-dimensional manifold is the constant vector

$$\mathbf{Y} = \begin{pmatrix} 0 \\ 1 \\ u_y \\ v_y \end{pmatrix}. \tag{3.21}$$

Note that the integrals are considered only in a small neighborhood *nbd* of the edge—that is, the neighborhood of the $x - y$ projection of edge E. This is because u_{xx} and v_{xx} are identically zero away from the edge neighborhood where the rapid variation occurs. First use the same approach as in the section on constant coefficient diffusion terms—that of assuming the edge E is aligned with the y axis and then rotating the results accordingly. We note that α^0 is equivalent to a function of y alone in an infinitesimal neighborhood of the edge E. Now take the integral $\int\int P\mathbf{L}\alpha^0 \mathrm{d}S$ over the neighborhood *nbd* to get

(3.22)
$$\begin{aligned}\int\int_{nbd} P\begin{pmatrix}0\\0\\u_{xx}\\v_{xx}\end{pmatrix}\alpha^0(y)\mathrm{d}S &= \int\int_{nbd}\sqrt{D}P\begin{pmatrix}0\\0\\a\\b\end{pmatrix}\sigma'(x)\alpha^0(y)\mathrm{d}x\mathrm{d}y\\ &= \int_{y=0}^{y=\ell_{01}}\left(\int_{x_{nbd}}\sqrt{D}P\begin{pmatrix}0\\0\\a\\b\end{pmatrix}\sigma'(x)\mathrm{d}x\right)\alpha^0(y)\mathrm{d}y\\ &= \frac{\ell_{01}}{2}\int_{\sigma=-1}^{\sigma=1}\sqrt{D(\sigma)}P(\sigma)\mathbf{A}\mathrm{d}\sigma,\end{aligned}$$

where ℓ_{01} is the length of the edge joining node 0 and node 1, as in Figure 3.2. Further, once the matrix $P(\sigma)$ has been postmultiplied through by the vector $\mathbf{A}$ (defined in (3.19)) and by $\sqrt{D(\sigma)}$, then

$$\sqrt{D(\sigma)}P(\sigma)\mathbf{A} = \frac{1}{\sqrt{D(\sigma)}}[D(\sigma)\mathbf{A} - D_1(\sigma)\mathbf{X}(\sigma) - D_2(\sigma)\mathbf{Y}], \tag{3.23}$$

where the vector $\mathbf{X}(\sigma)$ and variables $D(\sigma), D_1(\sigma)$, and $D_2(\sigma)$ are functions of σ:

$$\begin{aligned}\mathbf{X}(\sigma) &= \mathbf{M} + \mathbf{A}\sigma,\\ D(\sigma) &= |\mathbf{X}(\sigma)|^2|\mathbf{Y}|^2 - (\mathbf{X}(\sigma)\cdot\mathbf{Y})^2,\\ D_1(\sigma) &= (\mathbf{X}(\sigma)\cdot\mathbf{A})|\mathbf{Y}|^2 - (\mathbf{Y}\cdot\mathbf{A})(\mathbf{Y}\cdot\mathbf{X}(\sigma)),\\ D_2(\sigma) &= (\mathbf{Y}\cdot\mathbf{A})|\mathbf{X}(\sigma)|^2 - (\mathbf{X}\cdot\mathbf{A})(\mathbf{X}(\sigma)\cdot\mathbf{Y}).\end{aligned} \tag{3.24}$$

Simplifying, the expression $D(\sigma)\mathbf{A} - D_1(\sigma)\mathbf{X}(\sigma) - D_2(\sigma)\mathbf{Y}$ is linear in σ as quadratic terms cancel and may be written as $\mathbf{G} + \mathbf{H}\sigma$, where $\mathbf{G}$ and $\mathbf{H}$ are constant vectors in σ. Also notice that $D(\sigma)$ is quadratic in σ and may be written as $c_1 + c_2\sigma + c_3\sigma^2$, where c_1, c_2, and c_3 are constants in σ. Using this information, the integral $\int_{-1}^{1}\sqrt{D(\sigma)}P(\sigma)\mathbf{A}\mathrm{d}\sigma$ may thus be written as

$$\mathbf{G}\int_{-1}^{1}\frac{1}{\sqrt{c_1 + c_2\sigma + c_3\sigma^2}}\mathrm{d}\sigma + \mathbf{H}\int_{-1}^{1}\frac{\sigma}{\sqrt{c_1 + c_2\sigma + c_3\sigma^2}}\mathrm{d}\sigma. \tag{3.25}$$

In the one-dimensional theory of SGWMFE both the integrals multiplied by $\mathbf{G}$ and $\mathbf{H}$, respectively, were expressed analytically in [19] and [17], since they were necessary for the calculation of the contributions from the Laplacian terms in one dimension; however, sixteen point Gauss quadrature is used for the integrals for the studies carried out for this paper. Once the terms of the vector have been calculated, it is necessary to rotate the x and y components to the correct orientation, since the calculations were made based on the assumption that the $x-y$ projection of the edge E was aligned with the y axis.

This rotation is done by replacing the first two components of the vector (corresponding to its x and y components) and multiplying by a rotation matrix:

$$\begin{pmatrix}\dfrac{\mathrm{d}y}{\ell_i} & \dfrac{\mathrm{d}x}{\ell_i}\\ \dfrac{-\mathrm{d}x}{\ell_i} & \dfrac{\mathrm{d}y}{\ell_i}\end{pmatrix}. \tag{3.26}$$

3.5. Source terms. The contributions over a cell from a source term, say, $S_1(u,v)$ in L_1 onto its 0th node, are obtained as follows:

$$\int_T P \begin{pmatrix} 0 \\ 0 \\ S_1(u,v) \\ 0 \end{pmatrix} \alpha^0 \mathrm{d}S = \sqrt{D} P \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} \int_T S_1(u,v) \alpha^0 \mathrm{d}x\mathrm{d}y, \tag{3.27}$$

resulting in the equivalent compact form

$$\int_T S_1(u,v) \alpha^0 \mathrm{d}x\mathrm{d}y = A[\alpha^0 S_1(u,v)]_T. \tag{3.28}$$

3.6. Nonuniform diffusion terms in conservative form. Consider a nonconstant or nonlinear diffusion term of the form $L_1 = \nabla\cdot(a\nabla u)$, where $a = a(x,y,u,v)$. The contributions from this term in the cell onto its 0th node are obtained as follows:

$$\int_T P \begin{pmatrix} 0 \\ 0 \\ \nabla\cdot(a\nabla u) \\ 0 \end{pmatrix} \alpha^0 \mathrm{d}S = \int_T P \begin{pmatrix} 0 \\ 0 \\ a\nabla^2 u \\ 0 \end{pmatrix} \alpha^0 \mathrm{d}S + \int_T P \begin{pmatrix} 0 \\ 0 \\ \nabla a \cdot \nabla u \\ 0 \end{pmatrix} \alpha^0 \mathrm{d}S. \tag{3.29}$$

Consider the first integral on the right-hand side, noting that the term $\nabla^2 u$ is zero on each cell not including the edges. So the contributions from this first integral need only be considered in the δ-neighborhood of the edges in the support of α^0, as is the case for the constant coefficient Laplacian, as discussed in the section about constant coefficient diffusion terms. It will suffice to express the integral here for one edge E. Again we use the same approach as in the section on constant coefficient diffusion terms—that of assuming the edge E is aligned with the y axis and then rotating the results accordingly. As in [9] we note that, like α^0, $a(x,y,u,v)$ is equivalent to a function of y alone in an infinitesimal neighborhood of the edge E. Thus

$$\int_T P \begin{pmatrix} 0 \\ 0 \\ a\nabla^2 u \\ 0 \end{pmatrix} \alpha^0 \mathrm{d}S = \int_{y=0}^{y=\ell_{01}} \left(\int_{x_{nbd}} \sqrt{D} P \begin{pmatrix} 0 \\ 0 \\ \nabla^2 u \\ 0 \end{pmatrix} \mathrm{d}x \right) a(y) \alpha^0(y) \mathrm{d}y, \tag{3.30}$$

and the integral thus becomes (using the notation from (3.14))

$$\ell_{01}[a\alpha^0]_E \left(\int_{x_{nbd}} \sqrt{D} P \begin{pmatrix} 0 \\ 0 \\ \nabla^2 u \\ 0 \end{pmatrix} \mathrm{d}x \right). \tag{3.31}$$

The integral over the x_{nbd} is identical to the one derived for the Laplacian in (3.22), and thus the details will not be repeated here. The second integral in (3.29) can be treated as a source term:

$$\int P \begin{pmatrix} 0 \\ 0 \\ \nabla a \cdot \nabla u \\ 0 \end{pmatrix} \alpha^0 \mathrm{d}S. \tag{3.32}$$

Since P is constant over each cell and the term $\nabla a \cdot \nabla u$ is assumed bounded on each cell (u is constant on each cell and a is assumed continuous and bounded), then it is only necessary to consider the integral over each cell. For each cell T, use the theory derived for source terms with $S_1(u, v) = \nabla a \cdot \nabla u$, and similarly if there are equivalent terms in the second PDE $S_2(u, v) = \nabla a \cdot \nabla v$.

3.7. Regularization. As in the one-dimensional SGWMFE approach it may be necessary to add regularization terms in order to remove any degeneracies which may occur in the discrete system of ODEs. The regularization terms used here are similar to the terms introduced as pressure regularization in [8], as opposed to the internodal viscosity regularization terms more commonly known. As in the one-dimensional case, the need for regularization may arise when the solution has two consecutive cells with the same slope or when two nodes run into each other or overlap causing a degeneracy in the system of equations (see [17] for one-dimensional analysis). In order to remove the possibility of this happening, in [12] and [8] it was shown that adding regularization terms which are proportional to nodal velocities (described by Miller as internodal viscosities) is successful in two dimensions. These internodal viscosities are not needed for the porous medium equation; however, in some instances when the solution has an infinite slope it is necessary to use the pressure regularization terms as suggested by [13], added to keep nodes from overlapping.

The pressure terms used are derived from a quality functional. That is, the terms added are a constant times the negative gradient of a mesh quality functional Q, where the gradient is with respect to the discrete nodal velocities $[\dot{\mathrm{x}}_i, \dot{\mathrm{y}}_i, \dot{\mathrm{u}}_i, \dot{\mathrm{v}}_i]$. The quality functional used here is

$$Q = \frac{P^2}{\Delta S}, \tag{3.33}$$

on each cell T, where P is the perimeter of the triangular cell and ΔS is the area of the triangle in the four-dimensional space as introduced in (3.11). Thus for a four-dimensional set of ODEs resulting from the SGWMFE discretization, the pressure terms added to the equations at each node is

$$-C^2 \begin{pmatrix} \dfrac{\mathrm{d}Q}{\mathrm{d}x_i} \\ \dfrac{\mathrm{d}Q}{\mathrm{d}y_i} \\ \dfrac{\mathrm{d}Q}{\mathrm{d}u_i} \\ \dfrac{\mathrm{d}Q}{\mathrm{d}v_i} \end{pmatrix}. \tag{3.34}$$

The solutions are not very sensitive to the value of the pressure regularization parameter C^2, so long as C^2 is much smaller than the error tolerance, that is, so that the regularization terms added are well below the order of accuracy required. For the porous medium equation when the pressure terms are not needed, the addition of these terms makes no difference to the solution within the error tolerance. These terms are not needed for the case when $m = 1$ in the porous medium equation. However, as explained later, there are some cases when these terms are needed and the C^2 is set to $(tol/10)^2$, where tol is the local truncation error tolerance of the time integrator.

4. Results.

4.1. Porous medium equation. The parameter m in (1.1) determines the strength of the nonlinearity in the diffusion. In this paper we consider two cases, $m = 1$ and $m = 3$. For both cases the problem is started with an initial condition of $r_0 = 1/2$ and the initial profile is obtained from the similarity solution using this information. It is worthwhile noting that the choice of m is not limited; while not all possible integer values for $m > 0$ were tested, some integer values larger than 3 were tested, such as the problem with $m = 5$, which produced results in agreement with (1.2), as in the cases presented in this paper. The key point that makes the cases $m = 1$ and $m > 1$ different is that for the first case the slopes of the solutions are finite over the domain for this problem, and for cases $m > 1$ the solutions have infinite slopes at the moving boundary. In all cases, SGWMFE performs well.

The results illustrated in Figures 4.1 and 4.2 show the differences between the cases $m = 1$, which clearly diffuses faster, and $m = 3$, which has infinite slope near the moving boundary. The solutions were computed in the first quadrant using a two-dimensional SGWMFE code and the graphs show the solutions reflected onto the three remaining quadrants. This is because the problem is axially symmetric. The mesh plots are on the actual computational domain (see Figures 4.3 and 4.4), where it is seen that the SGWMFE code does indeed produce axially symmetric results. The speed of the spreading of the solutions can also be seen more clearly from the mesh plots. For both problems ($m = 1$ and $m = 3$) the initial conditions begin with an arced domain of radius 1/2. However, in the case $m = 1$, the solution boundary reaches a radius of almost 1.5, and in the case $m = 3$, the radius reaches a value of around 0.8 in the same amount of time. It is interesting that, regardless of the initial mesh chosen, for the porous medium equation, the meshes converge almost instantly, so that at any particular later time, the meshes will be the same; i.e., the mesh acts as an attractor. This can be seen in Figures 4.5 and 4.6.

It is worth mentioning that the case of $m = 3$ required use of pressure regularization terms. This is because the slopes are infinite and in the two-dimensional case (as in the one-dimensional case found in [17, 18]) the nodes tend to bunch up

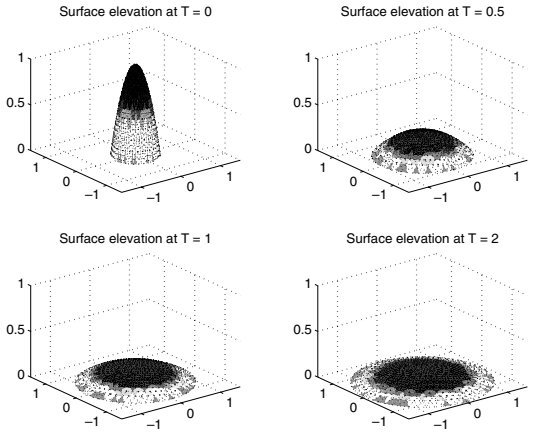

FIGURE 4.1. *SGWMFE solutions for the porous medium equation with* $m = 1$. *Solutions were obtained with* 136 *nodes initially distributed uniformly. Viscous* (A^2) *regularization terms and pressure* (C^2) *regularization terms were not needed for this problem.*

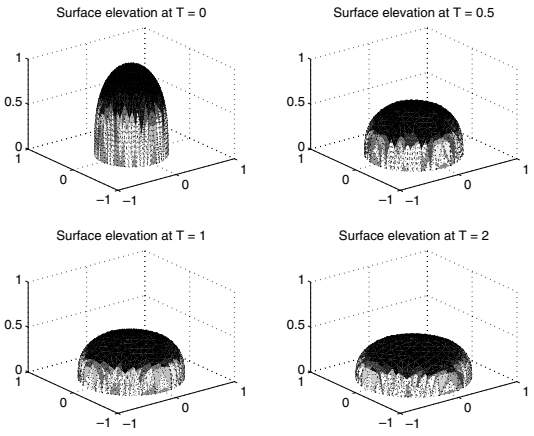

FIGURE 4.2. *SGWMFE solutions for the porous medium equation with* $m = 3$. *Solutions were obtained with* 136 *nodes initially distributed uniformly. Pressure regularization terms of* $C^2 = (10)^{-10}$ *were used for this problem. Viscous* (A^2) *regularization terms were not needed for this problem.*

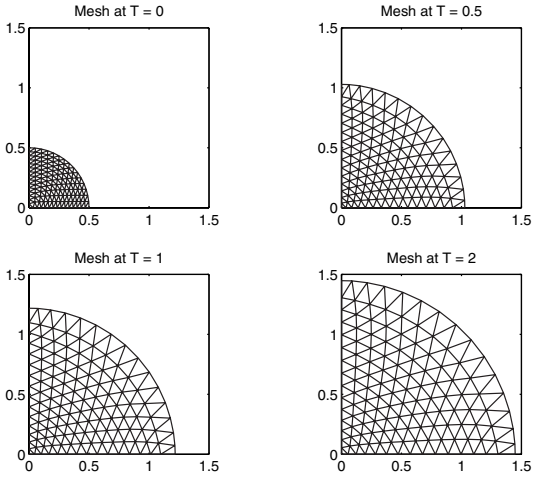

FIGURE 4.3. *SGWMFE meshes corresponding to solutions for the porous medium equation with* $m = 1$. *Solutions were obtained with* 136 *nodes initially distributed uniformly. Viscous* (A^2) *regularization terms and pressure* (C^2) *regularization terms were not needed for this problem.*

toward the infinite slope. In this case, it is so severe that the regularization pressure terms are needed in order to keep the nodes from overlapping and thus maintaining a well-defined mesh. The regularization pressure terms used for this problem are very small—much smaller than the time integrator tolerance used for this problem, which was chosen at 10^{-4}. Also this is the only problem presented in this paper where regularization terms were not needed, regardless of the value of m and regardless of the dimension in which the problem was solved. This is because the problem is diffusive and has its own natural diffusivity in the nonlinear terms, thus keeping the solution smooth. This is different from the shallow water equations because artificial viscosity was added for the purpose of maintaining a smooth solution in the numerical calculation.

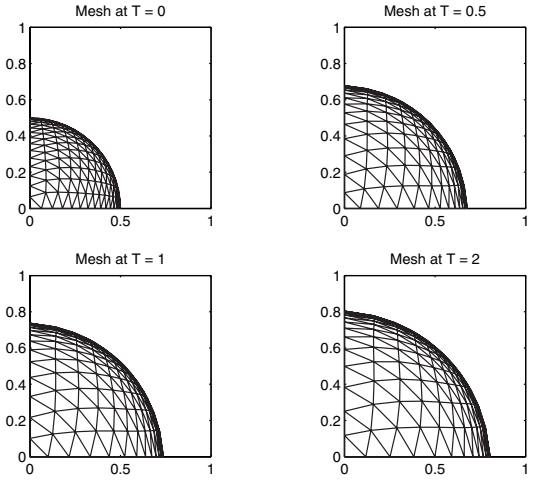

FIGURE 4.4. *SGWMFE meshes corresponding to solutions for the porous medium equation with* $m = 3$. *Solutions were obtained with* 136 *nodes initially distributed uniformly. Pressure regularization terms of* $C^2 = (10)^{-10}$ *were used for this problem. Viscous* (A^2) *regularization terms were not needed for this problem.*

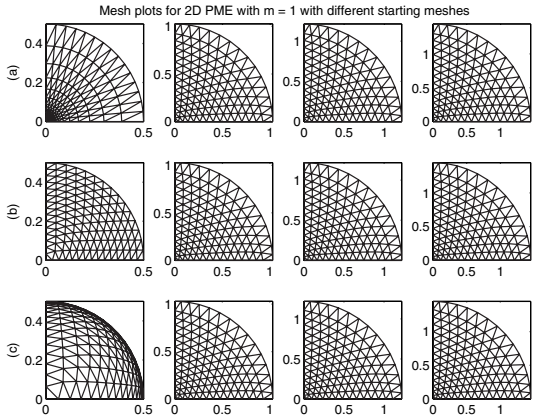

FIGURE 4.5. *SGWMFE meshes corresponding to solutions for the porous medium equation with* $m = 1$. *Solutions were obtained with* 136 *nodes at times* $T = 0, 0.5, 1, 2$ *shown from left to right. In* (a) *the initial mesh is one with nodes clustered toward the origin,* (b) *shows an initial mesh which is uniform, and* (c) *shows nodes initially clustered toward the moving boundary. Viscous* (A^2) *regularization terms and pressure* (C^2) *regularization terms were not needed for this problem.*

Convergence results for SGWMFE applied to the porous medium equation in two dimensions are shown. In the two-dimensional cases (regardless of the value of m) the slopes of the convergence lines for all the times shown are close to 1. Since the relation between N and the average mesh parameter h is $h = \sqrt{area(\Omega)/N}$, or $N \sim h^{-2}$ (where Ω is the area of the computational domain), this result corresponds to an h^2 rate of convergence, which is optimal for linear finite elements.

The L_∞ norm is chosen here for simplicity and to avoid introducing error from the numerical calculation of the L_1 norm over the nonuniform moving triangular grid. The L_∞ error norms are shown in Figures 4.7 and 4.8. As expected from the results shown in the one-dimensional cases in [17, 18], the convergence rates are the same

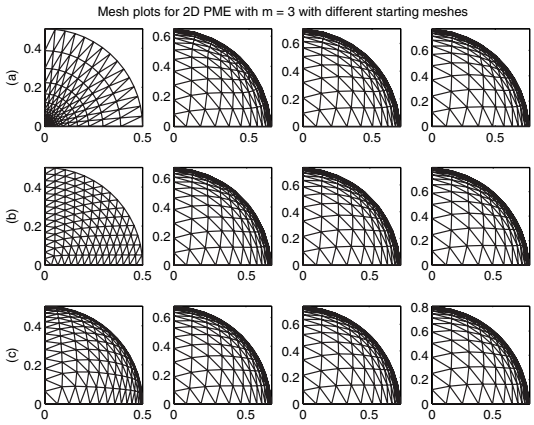

FIGURE 4.6. *SGWMFE meshes corresponding to solutions for the porous medium equation with* $m = 3$. *Solutions were obtained with* 136 *nodes at times* $T = 0, 0.5, 1, 2$ *shown from left to right. In* (a) *the initial mesh is one with nodes clustered toward the origin,* (b) *shows an initial mesh which is uniform, and* (c) *shows nodes initially clustered toward the moving boundary. Viscous* (A^2) *regularization terms were not needed for this problem. Pressure regularization terms of* $C^2 = (10)^{-10}$ *were used for this problem.*

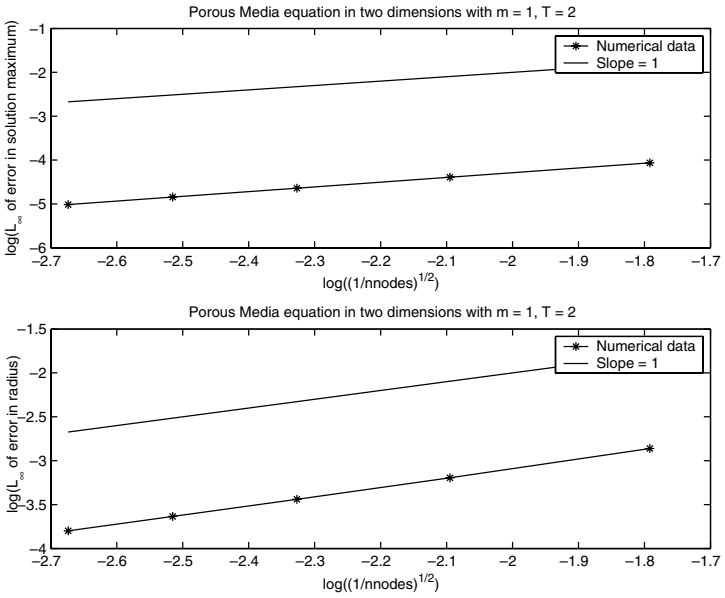

FIGURE 4.7. *Errors in SGWMFE solutions for the two-dimensional porous medium equation with parameter* $m = 1$. *The solutions were obtained using meshes with a varying number of nodes at time* $T = 2$. *Viscous* (A^2) *regularization terms and pressure* (C^2) *regularization terms were not needed for this problem. nnodes = number of nodes.*

regardless of the time to which the problem is solved. Thus only the convergence results for solutions calculated at time $T = 2$ are presented here.

The results in Figure 4.8 show more data results, from solving the problem using more nodes, than used to produce Figure 4.7 since originally it was not clear if the order of convergence of the radius was the same as the error over the entire solution

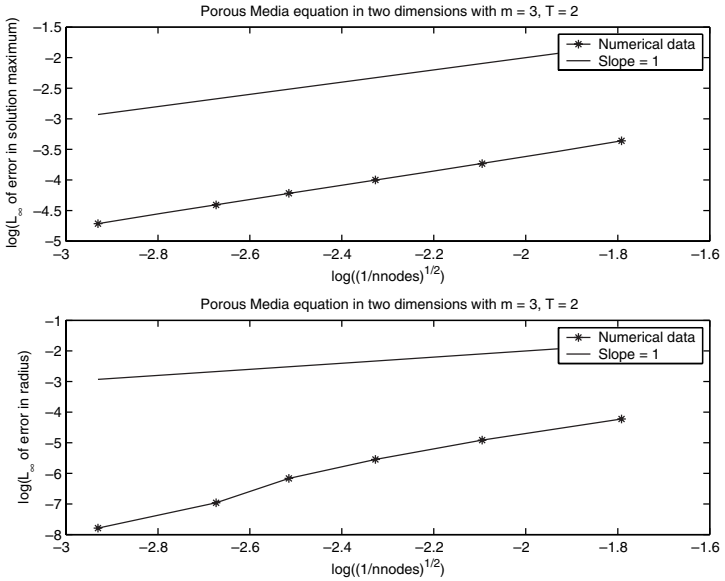

FIGURE 4.8. *Errors in SGWMFE solutions for the two-dimensional porous medium equation with parameter* $m = 3$. *The solutions were obtained using meshes with a varying number of nodes at time* $T = 2$. *Viscous* (A^2) *regularization terms were not needed for this problem and pressure regularization terms of* $C^2 = (10)^{-10}$ *were used for this problem. nnodes = number of nodes.*

domain, or if it was better for the case when $m = 3$. The extra data was needed since from the graph of the convergence rate, Figure 4.8, shows a higher order convergence rate than the case when $m = 1$ as the number of nodes are increased. However, since the error tolerance used for this problem is 10^{-4}, using more nodes than used to produce the convergence results will not be more conclusive, since the results shown are already of the same order as the error tolerance. If all the data were to be reproduced with a lower truncation error tolerance, this would lead to more accurate solutions, though the results would not necessarily show convergence rates that are significantly different.

4.2. Gray Scott equations. First we examine the convergence of the method on the Gray Scott equations. Since an analytic solution is not available, we construct a reference solution (to be regarded as the "exact solution") by using an extremely fine discretization—much finer than that used for the actual computation. Solutions with variable nodes are compared pointwise to solutions on a fine reference mesh with 55 by 55 nodes. We then define the error measure in the component u as

$$E = \| u_{ref} - u \| \, / \, \| u_{ref} \|, \tag{4.1}$$

where u is the actual SGWMFE solution, u_{ref} is the reference solution, and $\| \cdot \|$ is the discrete L_2 norm. Note that the values used for calculating the norms in (4.1) are mapped onto a fixed uniform fine mesh (using the linear shape functions) in order to enable objective comparison of the various meshes. Figure 4.9 shows the convergence curves of $E(N)$ on a log-log scale, where N is the total number of nodal points, showing that the solutions with a variable number of nodes converge to the solution on the finest mesh. Similar results have been obtained when applying this procedure to the shallow water equations.

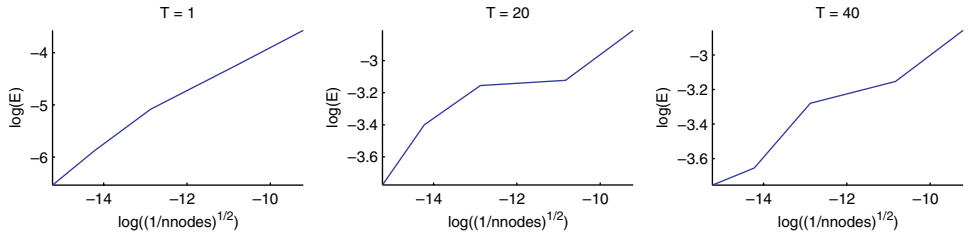

FIGURE 4.9. *SGWMFE convergence in u for the Gray Scott equations at times (from left to right): T = 1, 20, and 40. The error tolerance used is* 10^{-4}. *The corresponding viscous regularization terms of* $A^2 = 5(10)^{-8}$ *and pressure regularization terms of* $C^2 = (10)^{-8}$ *were used for this problem. nnodes = number of nodes.*

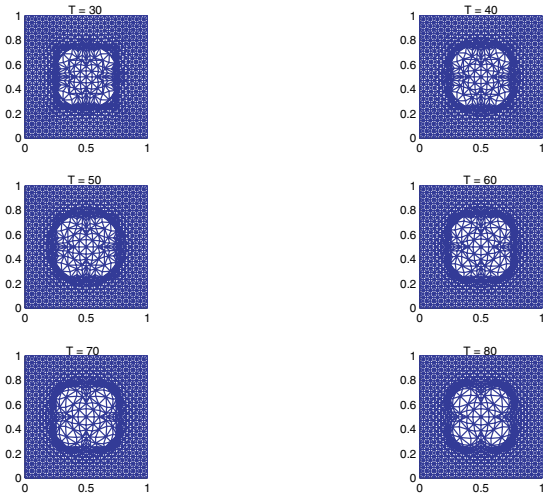

FIGURE 4.10. *SGWMFE meshes produced when solving the Gray Scott equations with* 35 *by* 35 *nodes. The error tolerance used is* 10^{-4}. *The corresponding viscous regularization terms of* $A^2 = 5(10)^{-8}$ *and pressure regularization terms of* $C^2 = (10)^{-8}$ *were used for this problem.*

Next we show mesh plots at several times for SGWMFE solutions of the Gray Scott equations using 35 by 35 nodes, much fewer nodes than for the solution plots following, for the sake of clarity of the figure. It is clear in Figure 4.10 that the mesh is attracted to the reaction fronts. See Figures 4.11 and 4.12 for contours of the solutions of the component u at corresponding times. The patterns observed in the solution figures are similar to those found in [20]. However, the meshes obtained for SGWMFE as compared to those shown in [20] are quite different, noting that for SGWMFE there is a much richer concentration of the nodes at the reaction front in comparison to the meshes produced by the method in [20].

4.3. Shallow water equations. Figure 4.13 shows SGWMFE results for the shallow water equations in two dimensions. The mesh plots are shown on the computational domain in order to make clearer comparisons between the mesh produced by SGWMFE and those produced by GWMFE. All of the numerical results for this problem conserve "mass" (the integral of the variable u over the domain) within the error tolerance. SGWMFE solutions are visually the same in comparison to the results obtained using GWMFE, though the meshes produced by each method differ.

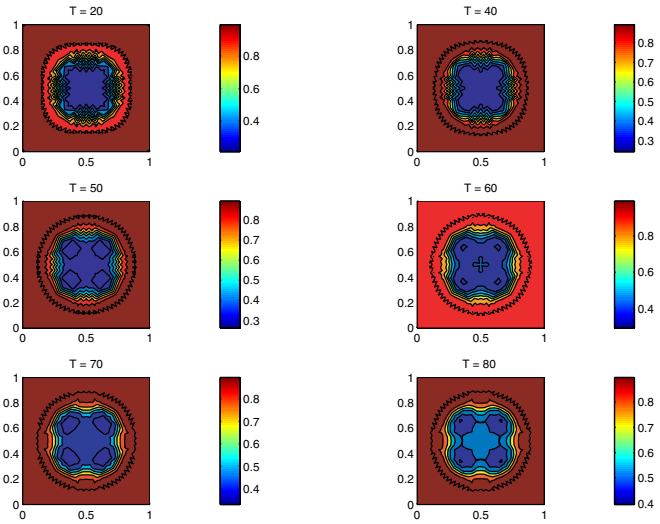

FIGURE 4.11. *SGWMFE solutions produced when solving the Gray Scott equations with* 45 *by* 45 *nodes. The error tolerance used is* 10^{-4}. *The corresponding viscous regularization terms of* $A^2 = 5(10)^{-8}$ *and pressure regularization terms of* $C^2 = (10)^{-8}$ *were used for this problem.*

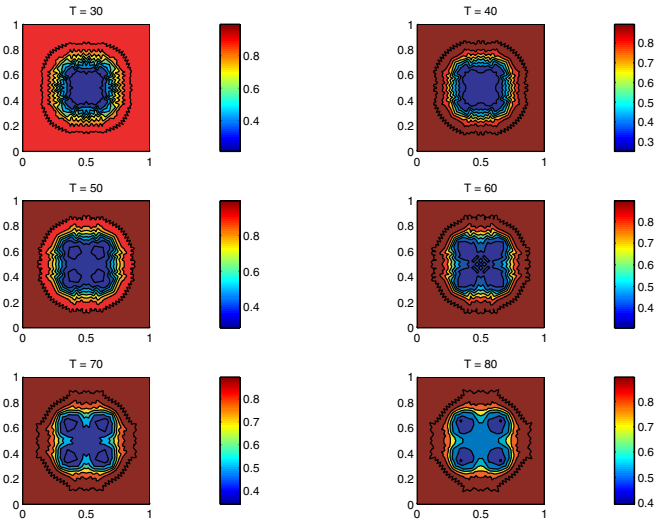

FIGURE 4.12. *SGWMFE solutions produced when solving the Gray Scott equations with* 55 *by* 55 *nodes. The error tolerance used is* 10^{-4}. *The corresponding viscous regularization terms of* $A^2 = 5(10)^{-8}$ *and pressure regularization terms of* $C^2 = (10)^{-8}$ *were used for this problem.*

The results shown here are calculated to the user-chosen tolerance, which was chosen to be 10^{-3} in this case. The solutions were in agreement with those obtained with the GWMFE code used in [9]. Further noting that the solutions are in agreement with GWMFE, it is worth discussing the uniqueness of the meshes produced. The meshes obtained for the shallow water equations are not "unique." In Figures 4.14 and 4.15, it is shown that with different initial meshes for solving the two-dimensional shallow water equations (with the same artificial viscosity), at various times, different

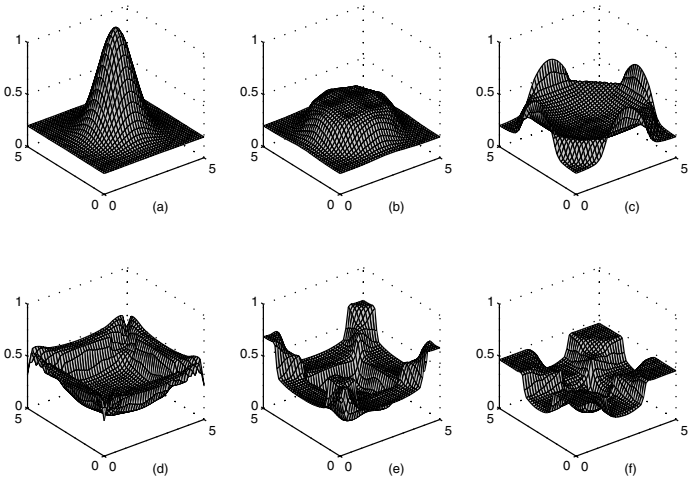

FIGURE 4.13. *SGWMFE solutions for the shallow water equations with artificial viscosity* $\epsilon = (10)^{-2}$ *at times* $T = 0, 1, 2, 3, 4, 5$. *Solutions were obtained with* 289 *nodes using the viscous regularization terms* $A^2 = 5(10)^{-5}$. *Regularization pressure terms were not needed for this problem.*

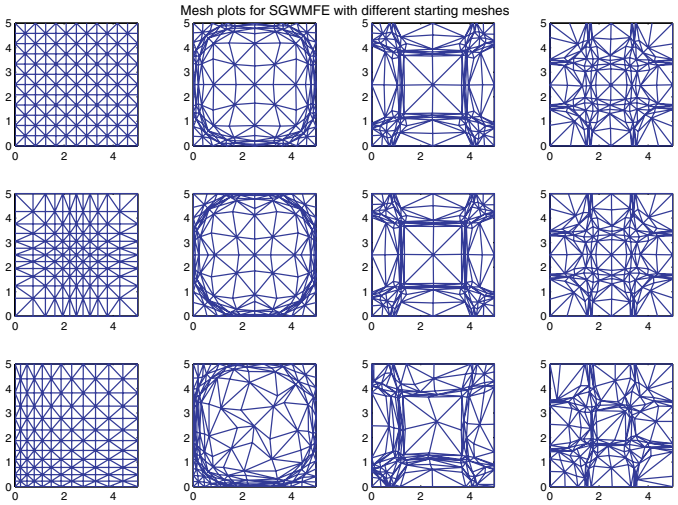

FIGURE 4.14. *SGWMFE meshes corresponding to solutions for the shallow water equations with artificial viscosity* $\epsilon = (10)^{-2}$ *at times* $T = 0, 2, 4, 5$. *Solutions were obtained with* 289 *nodes using the viscous regularization terms* $A^2 = 5(10)^{-5}$. *Pressure regularization terms were not needed for this problem.*

meshes are obtained (although the methods converge to solutions within the same accuracy, regardless of the initial mesh). It can also be seen in Figures 4.14 and 4.15, as expected, that the GWMFE and SGWMFE produce different meshes. This is true for both the one-dimensional and two-dimensional formulations of the shallow water equations. This could have been shown using results obtained for the one-dimensional shallow water equations; however, the results in two dimensions, presented here, are more demonstrative of the differences.

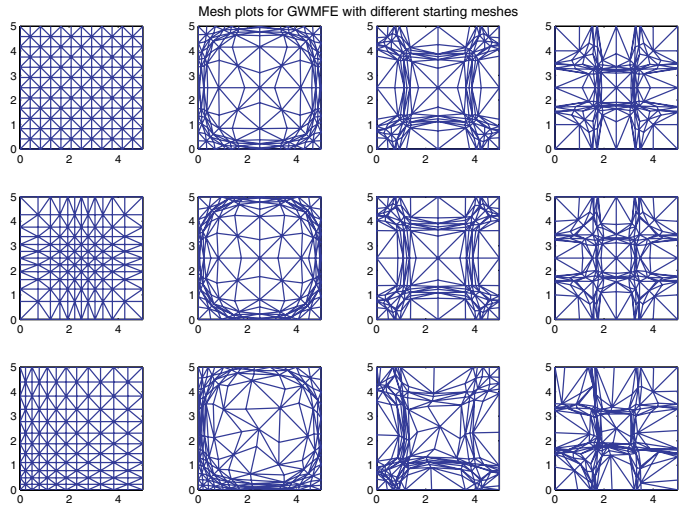

FIGURE 4.15. *GWMFE meshes corresponding to solutions for the shallow water equations with artificial viscosity* $\epsilon = (10)^{-2}$ *at times* $T = 0, 2, 4, 5$. *Solutions were obtained with* 289 *nodes using the viscous regularization terms* $A^2 = 5(10)^{-5}$. *Pressure regularization terms were not needed for this problem.*

5. Conclusions. The equations for SGWMFE have been presented using a projection matrix, making the underlying formulation easy to understand. The resulting equations resulting for SGWMFE are an elegant extension to existing gradient weighted methods. A convergence study of the method was carried out for the porous medium equation in two dimensions. The results show accurate solutions that correspond to an h^2 rate of convergence, which is optimal for linear finite elements. Further results were shown for the Gray Scott equations agreeing with results of a different moving mesh method and solutions of the nonlinear shallow water equations for both SGWMFE and GWMFE, obtaining equally accurate results on the different meshes produced by each corresponding method.

In [17] further calculations are given for the shallow water equations in both one and two dimensions, as well as for reaction diffusion equations in one dimension. In each case SGWMFE was straightforward to implement and gave reliable solutions for relatively little computational effort.

In conclusion, derivation and implementation of SGWMFE is straightforward and the formulation is easily generalized for arbitrary systems of PDEs in multiple dimensions. The method has proved to solve the problems presented here with steep moving fronts and a chemical reaction diffusion problem with pattern formation, both of which are difficult numerical problems. For such problems, SGWMFE can produce accurate results efficiently with respect to the number of nodes used as well as the calculation time.

6. Acknowledgments. This work has been motivated and to some extent, guided, by Keith Miller, whose pioneering work on gradient weighted methods contains the origins of SGWMFE, and we are immensely grateful to him for his help and advice. We are also very grateful to Andy Wathen and Mike Baines for discussions about moving finite element methods and to Mike Baines for suggesting a study of the porous medium equation.

REFERENCES

[1] M. Baines, *Moving Finite Elements*, Oxford University Press, New York, 1994.

[2] M. Baines, M. Hubbard, and P. Jimack, *A moving mesh finite element algorithm for the adaptive solution of time-dependent partial differential equations with moving boundaries*, Appl. Numer. Math., 54 (2005), pp. 450–469.

[3] R. E. Bank, W. C. Coughran, Jr., W. Fichtner, E. Grosse, D. Rose, and R. Smith, *Transient simulation of silicon devices and circuits*, IEEE Trans. CAD, 4 (1985), pp. 436–451.

[4] G. I. Barenblatt, *Scaling*, Cambridge Texts In Applied Mathematics, Cambridge University Press, Cambridge, UK, 2003.

[5] C. Budd, R. Carretero-Gonzlez, and R. Russell, *Precise computations of chemotactic collapse using moving mesh methods*, J. Comput. Phys., 202 (2005), pp. 463–487.

[6] C. J. Budd, G. Collins, W. Huang, and R. Russell, *Self-similar numerical solutions of the porous-medium equation using moving mesh methods*, Phil. Trans. Roy. Soc. London Ser. A, 357 (1999), pp. 1047–1077.

[7] C. J. Budd and M. Piggott, *The geometrix integration of scale-invariant ordinary and partial differential equations*, J. Comput. Appl. Math., 128 (2001), pp. 399–422.

[8] N. N. Carlson and K. Miller, *Design and application of a gradient-weighted moving finite element code* I: *In one dimension*, SIAM J. Sci. Comput., 19 (1998), pp. 728–765.

[9] N. N. Carlson and K. Miller, *Design and application of a gradient-weighted moving finite element code* II: *In two dimensions*, SIAM J. Sci. Comput., 19 (1998), pp. 766–798.

[10] J. L. Kyoung, W. McCormick, Q. Ouyang, and H. L. Swinney, *Pattern formation by interacting chemical fronts*, Science, 261 (1993), pp. 192–194.

[11] R. Li, W. Liu, and H. Ma, *Moving mesh method with error-estimator-based monitor and its applications to static obstacle problem*, J. Sci. Comput., 21 (2004), pp. 31–55.

[12] K. Miller, *A geometrical-mechanical interpretation of gradient-weighted moving finite elements*, SIAM J. Numer. Anal., 34 (1997), pp. 67–90.

[13] K. Miller, *private communication*, 2002.

[14] K. Miller and R. N. Miller, *Moving finite elements.* I, SIAM J. Numer. Anal., 18 (1981), pp. 1019–1032.

[15] Z. Tan, Z. Zhang, Y. Huang, and T. Tang, *Moving mesh methods with locally varying time steps*, J. Comput. Phys., 200 (2004), pp. 347–367.

[16] T. Tang, *Moving Mesh Methods for Computational Fluid Dynamics*, CSCAMM-05-04, Center for Scientific Computation and Mathematical Modeling, University of Maryland, College Park, MD, 2005.

[17] A. Wacher, *String Gradient Weighted Moving Finite Elements For Systems of Partial Differential Equations*, Ph.D. thesis, Oxford University, Oxford, UK, 2004.

[18] A. Wacher, *Mmpde vs. Sgwmfe, Experiments in One Dimension*, Tech. report, Oxford Computing Laboratory, Oxford, UK, 2005.

[19] A. Wacher, I. Sobey, and K. Miller, *String Gradient Weighted Moving Finite Elements for Systems of Partial Differential Equations*, Tech. report, Oxford Computing Laboratory, Oxford, UK, 2003.

[20] P. A. Zegeling and H. P. Kok, *Adaptive moving mesh computations for reaction–diffusion systems*, J. Comput. Appl. Math., 168 (2004), pp. 519–528.